



INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

MPR-based Pruning Techniques for Shortest Path Tree Computation

Juan Antonio Cordero,

N° 7329

Juin 2010

A large, light gray stylized 'R' logo that serves as a background for the text 'Rapport de recherche'.

***Rapport
de recherche***

MPR-based Pruning Techniques for Shortest Path Tree Computation

Juan Antonio Cordero*,

Thème : COM – Systèmes communicants
Équipe-Projet Hipercom

Rapport de recherche n° 7329 — Juin 2010 — 15 pages

Abstract: Multi-Point Relaying (MPR) is a well-known relay pruning algorithm that has proved to be useful for efficient dissemination in Mobile Ad hoc Networks (MANETs). But this technique may be useful for other tasks in MANET link-state routing as well. In particular, the approach is attractive for the selection of topology information to be flooded across the network. Requirements for such topology selection are however different from those applying for efficient dissemination, so approaches in such direction need to address these requirements and adapt or complement the MPR mechanism accordingly. This paper analyzes the main asymptotic properties of MPR and MPR-based topology selection algorithms, and provides sufficient conditions for the correctness of MPR-based topology selection. It examines as well in detail the MPR-based topology selection algorithm of MPR-OSPF, Path MPR, and shows that this algorithm may be unable, in certain conditions, to preserve optimal routes in its topology selection. The paper concludes by proposing and validating a modification of the Path MPR algorithm to overcome this sub-optimal performance.

Key-words: OSPF, OLSR, MANET, MPR, Routing, Path MPR, Graph, SPT

* INRIA Saclay / LIX, École Polytechnique, cordero@lix.polytechnique.fr

MPR-based Pruning Techniques for Shortest Path Tree Computation

Résumé : Multi-Point Relaying (MPR) est un algorithme de sélection de relais bien connu qui s'est avéré utile pour les tâches de diffusion efficace (*efficient flooding*) dans les réseaux mobiles ad hoc (MANET). Mais cette technique peut être utile aussi pour d'autres tâches en ce qui concerne le routage d'état-lien en réseaux MANET. En particulier, l'approche est intéressante pour la réduction des liens faisant partie de la topologie notifié à chaque routeur. Néanmoins, les conditions pour la sélection topologique sont assez différentes de celles à l'*efficient flooding*, et donc les approches basées sur MPR dans cette direction doivent répondre à ces conditions-là et s'adapter ou compléter le mécanisme de MPR en conséquence. Cet memorandum analyse les propriétés asymptotiques plus importantes de MPR et les algorithmes de sélection topologiques basées sur MPR, tout en fournissant des conditions suffisantes quant à l'exactitude des mécanismes de sélection topologique. Le memorandum examine aussi un algorithme concret de sélection topologique, *Path MPR*, et montre que cet algorithme, utilisé et spécifié dans l'extension d'OSPF pour MANET RFC 5449, ne peut pas assurer, en certaines conditions qui se détaillent, la préservation des routes optimales pour l'algorithme de sélection. Le memorandum conclut en proposant et validant une modification de l'algorithme *Path MPR* qui permet de corriger cet effet.

Mots-clés : OSPF, OLSR, MANET, MPR, Routing, Path MPR, Graph, SPT

Contents

1	Introduction	4
1.1	Terminology	4
2	Background	5
2.1	Multi-Point Relays – MPR	5
2.2	Requeriments of a Topology Pruning Algorithm	6
3	Using MPR for Topology Pruning	7
3.1	MPR Connection Analysis	7
3.2	MPR-based Topology Pruning Analysis	8
4	Path MPR as Topology Pruning Mechanism	9
4.1	The Path MPR Algorithm	9
4.2	Correctness in Unit Link Costs Scenarios	10
4.3	Correctness in Arbitrary Link Costs Scenarios	10
5	Fixing the Path MPR algorithm	11
5.1	Proposed Modification	11
5.2	Correctness and Characterization	12
5.3	Simulations	12
5.3.1	Configuration	12
5.3.2	Average Path Length	12
5.3.3	Average Number of Advertised Links	13
6	Conclusion	13
7	References	15

1 Introduction

The Multi-Point Relaying (MPR) technique is a well-known relay pruning algorithm. It has proved to be useful for performing efficient dissemination in networks characterized by bandwidth sharing and scarcity, such as Mobile Ad hoc Networks (MANETs) [6] [7] [8].

The MPR technique may be useful for other goals as well. There have been some extensions of the Multi-Point Relays algorithm, which are addressed to topology discovery in link-state routing protocols. In such protocols, topology information has to be flooded across the network so that every node keeps updated its local Link-State Database (LSDB) and thus compute optimal routes to all possible destinations. But in bandwidth-scarce networks as MANETs, such requirement of LSDB maintenance needs to be balanced with the minimization of control traffic, which includes the traffic dedicated to advertise the network topology. The MPR principle provides then a promising approach to reduce the set of links to advertise (pruning operation). Therefore, MPR-based topology pruning algorithms have been designed for OSPF MANET (*Path MPR* algorithm in MPR-OSPF [3]) and similar approaches are intended to be developed for OLSR [1] [5].

Requirements for such a topology pruning mechanism are nonetheless different to those applying for an efficient dissemination mechanism, as section 2 points out. They mostly relate to the quality of the resulting topology sub-graph. The paper discusses analytically in section 3 the fulfillment of such requirements both in the MPR technique and in the MPR-based mechanisms for topology pruning, and it proposes sufficient conditions for the asymptotic correctness of these mechanisms.

The paper then focus on a particular, standardized MPR-based topology extension, the Path MPR algorithm [3], and examine its performance in different network metric frameworks. Section 4 shows that the current specification of this algorithm may not provide, as claimed, shortest paths with respect to a non-unitary metric. Section 5 then proposes and formally validates an algorithm modification in order to assure its correctness as a topology pruning mechanism. The impact of such modifications is presented and discussed in the same section via simulations. Finally, section 6 concludes the paper.

1.1 Terminology

In the remainder of this paper, terms from graph theory (graph, vertex, edge) and from networking theory (network graph, node, link) are employed indistinctly to refer to the same concepts. Aside from that, the following terminology and notation conventions are used:

- A (network) graph is denoted by $G = (V, E)$, with $V = V(G)$ being the set of vertices and $E = E(G)$ the set of edges. Unless otherwise stated, it is assumed that G stands for a connected graph.
- Given a vertex (node) $x \in V$, $N(x)$ is the set of adjacent vertices (bidirectional neighbors) of x , $N_2(x)$ is the set of bidirectional 2-hop neighbors of x .

- Given two adjacent vertices (neighbor nodes) $x, y \in V$ and an edge metrics function $cost(e \in E) \in \mathbb{R}$, $cost(\overline{xy}) = cost(x, y)$ stands for the cost of the direct link \overline{xy} .
- Given two vertices (nodes) $x, y \in V$ and a path $p_{xy} = \{\overline{xm_1}, \overline{m_1m_2}, \dots, \overline{m_im_{i+1}}, \dots, \overline{m_ky}\}$ between x and y , $|p_{xy}| = k + 1$ is the length of p_{xy} , that is, the number of hops between x and y through the path p_{xy} . The shortest path (w.r.t. an edge metrics function $cost$) between x and y is denoted by p_{xy}^* . It will be considered as well an extended path metrics function $cost(p_{xy})$, defined as

$$cost(p_{xy}) = cost(x, m_1) + \sum_{i=1}^{k-1} cost(m_i, m_{i+1}) + cost(m_k, y)$$

- Given two vertices (nodes) $x, y \in V$, $dist(x, y)$ is the cost of the optimal path (w.r.t. $cost$) between x and y , that is $dist(x, y) = cost(p_{xy}^*)$. Similarly, given two vertices $x, y \in V$ reachable in 2 hops, it will be denoted by $dist_2(x, y)$ the cost of the optimal path between x and y in 2 hops or less (so called *local shortest path*).

2 Background

This section briefly describes the basics about the MPR technique, as well as the requirements that a topology pruning algorithm based on local node decision should fulfill in order to enable valid Shortest Path Tree (SPT) computation.

2.1 Multi-Point Relays – MPR

The Multi-Point Relays (MPR) [6] technique is an algorithm that enables a node to select a subset of its 1-hop neighbors, so-called *multi-point relays*, such that each 2-hop neighbor is reachable through (at least) one of the selected 1-hop neighbors. Such condition is known as the *MPR coverage criterion*, and can only be achieved if nodes are aware of their 2-hop neighbors – *e.g.* by exchange of Hello messages [3]. Figure 1 illustrates the benefits of the MPR technique for flooding optimization, with respect to the classic flooding consisting of allowing every neighbor to retransmit any packet coming from the source.

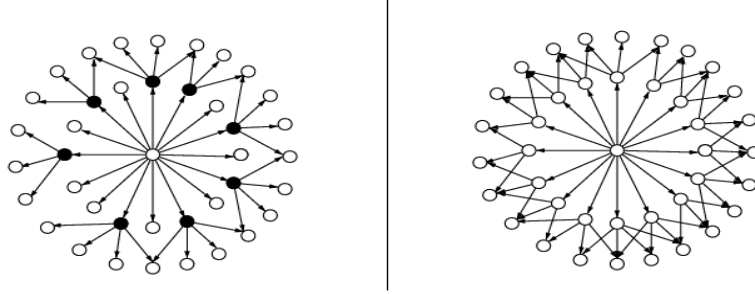


Figure 1: **(a)** Multi-Point Relays flooding *vs.* **(b)** classic flooding. *Solid balls represent neighbors selected as multi-point relays.*

Several heuristics are possible as long as they satisfy the *MPR coverage criterion*. In this paper it is used the following one, inspired in [3]:

1. **Input:** $x, N(x) = N, N_2(x) = N_2$.
2. $MPR = \{\emptyset\}$
3. $MPR \leftarrow \{n \in N : \exists m \in N_2, m \text{ is only covered by } n\}$
4. while $(\exists \text{ uncovered } m \in N_2)$
 $MPR \leftarrow n \in N : \text{covers max.\# of uncovered } m \in N_2$
5. **Output:** $MPR(x, N, N_2)$

2.2 Requiriments of a Topology Pruning Algorithm

Formally, a *topology pruning algorithm* is an algorithm that extracts, from a given network graph $G = (V, E)$, a subgraph $G' = (V, E') \subseteq G$ containing the same set of vertices and a subset of the edges (links) of the original network graph (see figure 2).

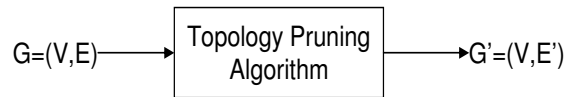


Figure 2: A topology pruning algorithm.

Such subgraph G' has to be computed locally (by every node in the network) and the resulting output is to be used as a basis for the Shortest Path Tree (SPT) calculation of the corresponding node. Hence, this subgraph needs to fulfill the following requirements:

- **Connection.** A non-connected subgraph would not permit the computing node to reach destinations in connected components other than the own.
- **Preservation of network-wide shortest paths.** SPT calculation algorithms (such as Dijkstra or Bellman-Ford) over G' identify the optimal paths in G' w.r.t. a metric. These optimal paths will correspond to those of G if and only if the G -shortest paths are in G' .

3 Using MPR for Topology Pruning

This section explores the above-mentioned requirements in subgraphs generated by the MPR selection algorithm and the MPR-based topology pruning algorithms. Subsection 3.1 focus on MPR connection issues, whereas subsection 3.2 elaborates on connection and shortest paths preservation of MPR-based topology pruning algorithms.

3.1 MPR Connection Analysis

Given a network graph $G = (V, E)$, let us consider the subgraph $S_{MPR} = (V, E_{MPR})$, hereafter called *MPR graph*, with $E_{MPR} = \{\overline{xy} : x \in V, y \in MPR(x)\} \subseteq E$ being the set of links between nodes from the network and their MPRs. It is immediate to show that the MPR graph is dense (meaning that every vertex in the network graph is either connected or at distance 1 of the MPR graph) [8], but it is not necessarily connected. Figure 3.a shows an example of disconnection in a 3-diameter network. Note that the displayed MPR election is legitimate since it satisfies the MPR coverage criterion.

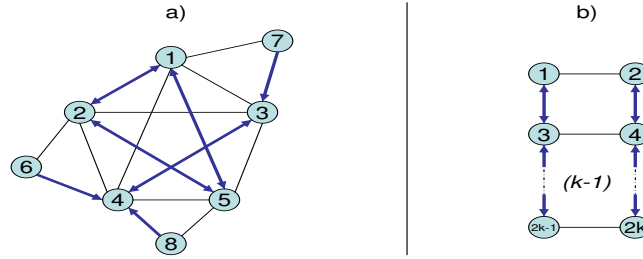


Figure 3: **(a)** Disconnection of the MPR set. *Thick directed lines represent MPR selection relationships.* **(b)** Disconnection of the MPR set in a k -diameter network.

More in general, for every value $k > 0$, it is easy to find a k -diameter network in which there are valid non-connected MPR set configurations (see figure 3.b).

This fact prevents MPR as-is to be used as an algorithm for construction of subgraphs that need to be connected, in particular subgraphs to be used for Shortest Path Tree (SPT) computation in a link-state routing protocol. A subgraph solely including MPR links might be unable to reach all destinations in the network.

Attempts to use the MPR as an algorithm for building such subgraphs, *e.g.* [3], are thus required to include additional links in the MPR subgraph in order to assure the connectivity of the resulting subgraph. The following two *Lemmas* prove that the addition of links from a single node of the network to all its neighbors is sufficient to guarantee such connection: *Lemma 1* shows that connected components of the MPR graph are dense and *Lemma 2* concludes that the union of the MPR graph and the set of links from a single node to its neighbors is necessarily connected.

Lemma 1 *Let $G = (V, E)$ be a network connected graph, and $H \subseteq G$ the subgraph of G containing the links from every vertex in the graph to all its MPRs. Then, every connected component of H is dense over G .*

Proof: Let $H^{cx} \subseteq H$ be a connected component of H . Let us consider $x \in H^{cx}$. It will be proved, by induction over k , that every vertex $z \in G$ at a distance k (in hops, $k < \infty$ because G is connected) from x has (at least) a neighbor that belongs to H^{cx} .

- $k = 1$ is trivial from the definition.
- $k = 2$, then z is a 2-hop neighbor of x and, by definition of the MPR, there will be a vertex $y \in N(x) \cap N(z)$ so that $\overline{xy} \in H^{cx}$.
- $k \implies k+1$. Let us consider the vertex $y \in G$ satisfying $\text{dist}(x, y) = k$, $y \in N(z)$. Note that such node y exists because $\text{dist}(x, z) = k + 1$, and by induction hypothesis, y is at a distance ≤ 1 from H^{cx} . Let t be the closest node of H^{cx} to y . Then, t is either a neighbor or a 2-hop neighbor of z ; in both cases, the argument for $k = 1, 2$ concludes that the $\text{dist}(z, H^{cx}) \leq 1$, and thus H^{cx} (and, more in general, every connected component of H) is dense in G .

Lemma 2 *Let $G = (V, E)$ be a network connected graph, and $H \subseteq G$ the subgraph of G consisting of:*

1. $H_1 \subseteq G$: *For every vertex $x \in V$, the edges from x to the neighbor vertices selected by x as MPRs.*
2. $H_2 \subseteq G$: *For a certain $s \in V$, the edges from s to every neighbor of s .*

Then, H is connected.

Proof: It is known from *Lemma 1* that, in case that there are several connected components of H_1 (that is, H_1 is disconnected), all of them are dense over G , meaning that every vertex of G has at least a neighbor belonging to each of them. Then, the subgraph that results from adding the links from any vertex of G (say $s \in G$) to all its neighbors (H_2) to H_1 will necessarily be connected. Note that the argument is valid with an arbitrary s .

Note that these results also imply that the number of connected components of the MPR graph is upper-bounded by the minimum number of neighbors achieved by a node within the network.

3.2 MPR-based Topology Pruning Analysis

Let us consider a *MPR-based topology pruning algorithm*, denoting by such a distributed topology pruning algorithm that consists of running MPR in every node of the network. Such algorithm computes its output among a subgraph of its 2-hop neighborhood, and it has to provide routes to every 2-hop neighbor. Then, the above-presented properties for the MPR graph apply as well to the subgraph generated by such MPR-based topology pruning algorithm. In particular, such subgraph is not necessarily connected (and thus may be unable to provide shortest paths to all destinations).

According to *Lemma 2*, the subgraph generated by a MPR-based topology pruning algorithm is connected if it is joined by the links of any node to its 1-hop neighbors. In particular, for a node $x \in V$ (being $G = (V, E)$ a network connected graph), the subgraph resulting from the union of links from x to its 1-hop neighbors and the subgraph generated by an MPR-based topology pruning algorithm is connected. If such algorithm preserves local the shortest paths from the 2-hop neighbors of x to x , then the resulting subgraph contains the shortest paths to all destinations in the network (*Lemma 3*).

Lemma 3 *Let $G = (V, E)$ be a network graph, an edge metrics function $\text{cost}(e \in E)$, a node $s \in V$ and a subgraph $G'_s = (V, E'_s)$ including:*

1. the edges connecting s to its 1-hop neighbors, and
2. for every node x of the network, the edges from x to those 1-hop neighbors of x providing local shortest paths from every 2-hop neighbor of x to x .

Then, the Dijkstra algorithm computed on a source node s over G'_s selects the shortest paths in G from the source to every possible destination.

Proof: Since the Dijkstra algorithm selects the shortest paths of the graph (w.r.t. a given metrics *cost*) over which it is computed, it needs to be proved that the shortest paths from s in G are contained in G'_s , i.e., $SPT_s(G) \subset G'_s \subset G$. Let z be an arbitrary node $z \in V$, $\overline{s z}_{sh-p}$ be the shortest path (w.r.t. *cost*) between s and z , and let $d(x, y)$ be the distance in hops between x and y .

- If $d(s, z) = 1$, $\overline{s z}_{sh-p} \in G'$ by condition 1 of the hypothesis.
- For $d(s, z) = n > 1$, let $\{m_i\}$ be the intermediate nodes of $\overline{s z}_{sh-p}$, so that $d(s, m_i) = i$. The edge $\overline{s m_1}$ belongs to G'_s by definition of G'_s (condition 1). The edge $\overline{m_i m_{i+1}}$ (consider $\overline{m_1 z}$ if $n = 2$) is included in G'_s because m_i is part of the local shortest path from s (2-hop neighbor of m_{i+1}) to m_{i+1} (condition 2 of the hypothesis about G'_s). Repeating the argument along $\overline{s z}_{sh-p}$ for $\{m_j\}_{1 \leq j < n}$, leads to the conclusion that all segments $\overline{s m_1}, \dots, \overline{m_i m_{i+1}}, \dots, \overline{m_{n-1} z}$ belong to G'_s and thus $\overline{s z}_{sh-p}$ belongs too.

4 Path MPR as Topology Pruning Mechanism

This section focus on a particular case of MPR-based topology pruning algorithm – the Path MPR algorithm, used in the standard OSPF MANET extension and specified in Appendix B of [3]. Such algorithm is described in subsection 4.1, and its correctness in terms of preservation of shortest paths is discussed in subsections 4.2 and 4.3, for unit-cost and arbitrary link cost scenarios, respectively.

4.1 The Path MPR Algorithm

According to the specification, the Path MPR algorithm intends to "provide the router with a Path-MPR set (...) such that for any element of N or N_2 that is not in the Path-MPR set, there exists a shortest path that goes from this element to the router through a neighbor selected as Path-MPR (unless the shortest path is only one hop)" [3]. The subgraph generated by Path MPR selection in every node of the network should thus include, for any node x of the network, the links to x from the neighbors providing local shortest paths (w.r.t. a given *cost* function) from the 2-hop neighborhood of x to x . Note that these links are directed, meaning that the Path MPR supports links with different costs depending on the direction.

The Path MPR algorithm extracts from the set of 1-hop neighbors of the computing node x a subset of neighbors (so called $N'(x)$) for which the direct link to x is as well the local shortest path w.r.t. the current metric. The algorithm extracts as well from the set of 2-hop and 1-hop neighbors of x a subset of neighbors (so called $N'_2(x)$) for which the local shortest path is not direct (it has 2 hops). Then, it computes the MPR algorithm from x (see subsection 3.1) over the 2-hop neighborhood subgraph resulting of considering $N'(x)$ as 1-hop neighborhood and $N'_2(x)$ as 2-hop neighborhood. The algorithm can be thus summarized as follows:

1. **Input:** $x, N(x), N_2(x)$.
2. The following subsets, $N' \subseteq N, N'_2 \subseteq N \cup N_2$, are calculated:

$$\begin{aligned} N' &= \{n \in N | \text{cost}(x, n) = \text{dist}_2(x, n)\} \\ N'_2 &= \{n \in N, N_2 | n \notin N', \exists m \in N' : \\ &\quad : \text{cost}(n, m) + \text{cost}(m, x) = \text{dist}_2(n, x)\} \end{aligned}$$

3. The router runs the MPR selection procedure with arguments $x, N'(x)$ and $N'_2(x)$.
4. **Output:** $\text{PathMPR}(x, N, N_2) = \text{MPR}(x, N', N'_2)$

Note that the Path MPR algorithm is a MPR-based topology pruning algorithm in the sense of subsection 3.2; and therefore the results presented in that section apply.

4.2 Correctness in Unit Link Costs Scenarios

Let us assume that the network links have a uniform cost. Then, the sets $N'(x)$ and $N'_2(x)$ computed by the Path MPR algorithm from a node x are expressed as follows:

$$\begin{aligned} N' &= \{n \in N | \text{cost}(x, n) = \text{dist}_2(x, n)\} = \\ &= [\text{cost}(x, n) = \text{dist}_2(x, n) = 1] = N \\ N'_2 &= \{n \in N \cup N_2 | n \notin N', \exists m \in N' : \\ &\quad : \text{cost}(n, m) + \text{cost}(m, x) = \text{dist}_2(n, x)\} = \\ &= [N = N'] = \{n \in N_2 | \exists m \in N : \\ &\quad : \text{cost}(n, m) + \text{cost}(m, x) = \text{dist}_2(n, x)\} = \\ &= [\text{dist}_2(n, x) = \text{cost}(n, m) + \text{cost}(m, x) = 2] = N_2 \end{aligned}$$

Thus, the Path MPR algorithm becomes equivalent to MPR. Neighbors selected as Path MPRs of a source x provide coverage to/from every 2-hop neighbor of x . Since all paths from 2-hop neighbors to x have a cost 2 (number of hops), that trivially means that the Path MPR algorithm provides local shortest paths in unit link cost networks.

4.3 Correctness in Arbitrary Link Costs Scenarios

In case that the link costs can take arbitrary values, the Path MPR algorithm may be unable to preserve shortest paths 2 hops away from the computing node. Figure 4 illustrates a simple example in which the Path MPR algorithm computed on node 1 selects a neighbor not providing local shortest paths from the 2-hop neighbors of 1 to 1.

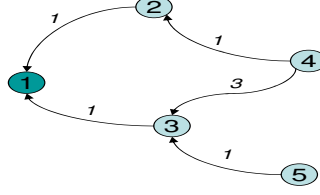


Figure 4: Path MPR malfunctioning example, w.r.t. node (1).

It is immediate to check that, for this case, the sets $N(1)$ and $N'_2(1)$ have the following composition:

$$\begin{aligned} N'(1) &= \{n \in N(1) : \text{cost}(n, 1) = \text{dist}_2(n, 1)\} = \{2, 3\} \\ N'_2(1) &= \{m \in N(1) \cup N_2(1) : \exists n \in N'(1) : \\ &\quad : \text{cost}(m, n) + \text{cost}(n, 1) = \text{dist}_2(n, 1)\} = \{4, 5\} \end{aligned}$$

Thus, according to the algorithm presented in subsection 4.1, the output from the Path MPR selection would be $\text{PathMPR}(1) = \{3\}$, since node (3) would be sufficient for covering all nodes in $N'_2(1)$ (MPR coverage criterion). This election would nonetheless not contain the shortest path from (4) to (1), $p_{41}^* = \{42, 21\}$.

In general terms, the problem shown in figure 4 comes from the fact that MPR is a cost-agnostic algorithm that only relies on coverage, while the Path MPR algorithm is expected to select links according to cost minimization rules. By running MPR over the subgraph formed by $N'(x)$ and $N'_2(x)$, the algorithm may select vertices of $N'(x)$ providing sub-optimal paths (in terms of *cost*) from $N'_2(x)$ to x , if they provide a better coverage (in terms of number of covered vertices belonging to $N'_2(x)$) than the vertices providing optimal (local shortest) paths.

5 Fixing the Path MPR algorithm

This section proposes a modification on the Path MPR algorithm in order to avoid non-optimal paths selection phenomena such as those observed in figure 4. The correction is described and analytically justified in subsections 5.1 and 5.2, whereas subsection 5.3 presents some simulations results that illustrate the impact of such modification in the algorithm performance.

5.1 Proposed Modification

If the Path MPR algorithm relies on MPR to select the relays providing shortest paths from 2-hops away to the source, it needs to exclude not only those nodes not providing sub-optimal paths ($N'(x) \setminus N(x) = \{n \in N(x) : \text{cost}(x, n) > \text{dist}_2(x, n)\}$), but also sub-optimal links (in terms of cost) from the subgraph over which MPR runs.

Therefore, the subgraph $S'_x \subseteq G$ over which MPR should be run for selecting the Path MPRs of a node x has the following expression:

$$\begin{aligned}
V(S'_x) &= N'(x) \cup N'_2(x) \\
E(S'_x) &= \{\overline{xn} \in E(G) : n \in N'(x)\} \cup \\
&\quad \{\overline{nm} \in E(G) : n \in N'(x), m \in N'_2(x), \\
&\quad \text{cost}(\{\overline{xn}, \overline{nm}\}) = \text{dist}_2(x, m)\}
\end{aligned}$$

5.2 Correctness and Characterization

By construction of the subgraph S'_x , if a path $p_{xz} = \{\overline{xy}, \overline{yz}\}$ is not optimal, with $y \in N'(x), z \in N'_2(x)$, then \overline{yz} will not belong to $E(S'_x)$. That assures that this modification of the Path MPR algorithm is able to select the local (2 hops) shortest paths to the computing node x . Thus, *Lemma 3* applies. That proves the following *Lemma 4*:

Lemma 4 *Let $G = (V, E)$ be a network graph. Given a vertex $x \in V$ representing a node of the network, let us consider the subgraph $S_x \subseteq G$ which results from joining:*

1. *the subgraph generated by the Path MPR links of every node in the network, and*
2. *the links from x to its 1-hop neighbors $n \in N(x)$.*

Then, S_x contains the network-wide shortest paths from x to every other destination in the network.

5.3 Simulations

The following simulation results illustrate the impact of the proposed modification on the Path MPR algorithm.

5.3.1 Configuration

The presented results correspond to the simulation in Maple of ideal, static and connected networks in which links are bidirectional.

Nodes are distributed uniformly through a fixed square grid ($400 \times 400m$), and have a uniform radio range $r = 150m$. Link costs satisfy $\text{cost}(e) \in \mathbb{N}, 1 \leq \text{cost}(e) \leq 10$. Results are presented for a random link cost model ($\text{cost} \sim \text{Uniform}(1, 10)$) and for a distance-based link cost model ($\text{cost}(\overline{xy}) = \lfloor 10^{\frac{\text{dist}(x,y)}{r}} \rfloor + 1$). Each value corresponds to the mean value of 20 samples.

5.3.2 Average Path Length

The results show that the average length of paths computed from a fixed node id (chosen 1, without loss of generality) to every other destination in the network. It can be observed (fig. 5) that the modified algorithm produces significantly shorter (in terms of *cost*) paths than the sub-optimal algorithm, not only with random costs links but also with more realistic distance-based metrics.

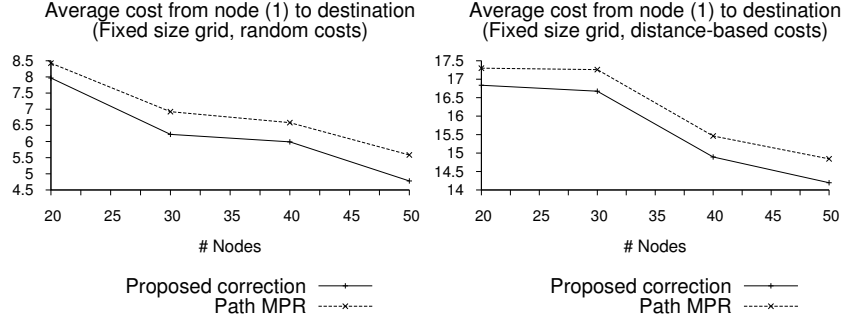


Figure 5: Average cost to destination from node (1) with (a) a random cost matrix, (b) a distance-based cost matrix.

This path cost reduction is at the expense of a slightly higher average number of hops (and thus of retransmissions). This effect is due to the fact that the modified algorithm runs MPR on a 2-hop neighborhood subgraph in which some additional links have been excluded (see subsection 5.1), which causes a larger or equal number of (cheaper) hops.

5.3.3 Average Number of Advertised Links

The exclusion of non-optimal links in the 2-hop neighborhood subgraph leads as well to a bigger number of advertised links (fig. 6). Since relays are no longer selected according to their number of covered 2-hop neighbors, the average number of covered neighbors per Path MPR drops necessarily, meaning that more Path MPRs are needed for the same coverage.

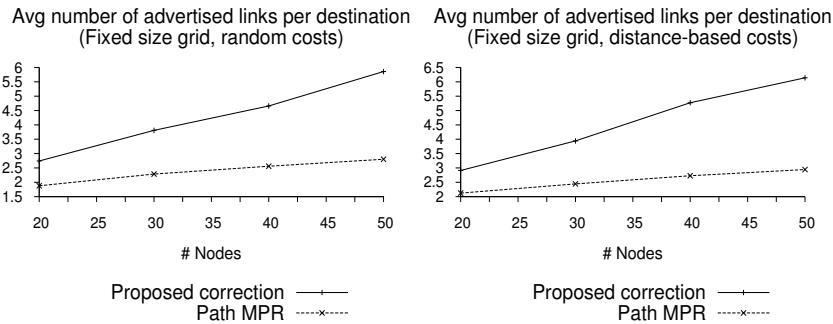


Figure 6: Average number of advertised PMPR links per node, with (a) a random cost matrix, (b) a distance-based cost matrix.

6 Conclusion

This paper has explored the use of MPR as a basis for topology pruning algorithms for MANETs, addressed to reduce the amount of topology information required for enabling every node to build its SPT. MPR properties makes it appropriate for optimizing the control traffic dedicated to topology diffusion while preserving essential information for the nodes. But it needs to be adapted in

order to fulfill the various requirements that such a topology pruning algorithm imposes. This paper has presented several analytical results in this domain, both characterizing inner asymptotic properties of MPR and providing sufficient conditions for the correct operation of an MPR-based topology pruning algorithm.

The paper has examined as well the correctness of the Path MPR algorithm in various link cost models. It has shown that such algorithm may lead, in asymptotic conditions, to sub-optimal relay elections which may prevent nodes to select valid shortest paths. Finally, it has thus proposed and validated formally a modification that overcomes this sub-optimal behavior.

7 References

- [1] C. Dearlove, T. Clausen, P. Jacquet: *The Optimized Link State Routing Protocol version 2*. Internet-Draft, `draft-ietf-manet-olsrv2-11.txt`. IETF. April 2010. (*work in progress*)
- [2] J. A. Cordero: RR 6827, *On MPR-OSPF Specification and Implementation in Quagga/GTNetS*. INRIA Research Report. February 2009.
- [3] E. Baccelli, P. Jacquet, D. Nguyen, T. Clausen: RFC 5449, *OSPF Multipoint Relay (MPR) Extension for Ad Hoc Networks*. IETF. February 2009.
- [4] J. A. Cordero: Master Thesis. *Evaluation of OSPF Extensions in MANET Routing*. Équipe HIPERCOM, Laboratoire d'Informatique (LIX), École Polytechnique. September 2007.
- [5] T. Clausen, P. Jacquet: RFC 3626, *Optimized Link State Routing Protocol (OLSR)*. IETF. October 2003.
- [6] A. Qayyum, L. Viennot, A. Laouiti: *Multipoint Relaying for Flooding Broadcast Messages in Mobile Wireless Networks*. Hawaii International Conference in System Sciences (HICSS-35), 2002.
- [7] P. Jacquet, A. Laouiti, P. Minet, L. Viennot: "Performance of Multipoint Relaying in Ad Hoc Mobile Routing Protocols", in E. Gregori *et al.*: *Networking 2002, LNCS 2345*, 2002.
- [8] C. Adjih, P. Jacquet, L. Viennot: RR 4597, *Computing Connected Dominated Sets with Multipoint Relays*. INRIA Research Report. October 2002.



Centre de recherche INRIA Saclay – Île-de-France
Parc Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 Orsay Cedex (France)

Centre de recherche INRIA Bordeaux – Sud Ouest : Domaine Universitaire - 351, cours de la Libération - 33405 Talence Cedex
Centre de recherche INRIA Grenoble – Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier
Centre de recherche INRIA Lille – Nord Europe : Parc Scientifique de la Haute Borne - 40, avenue Halley - 59650 Villeneuve d'Ascq
Centre de recherche INRIA Nancy – Grand Est : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex
Centre de recherche INRIA Paris – Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex
Centre de recherche INRIA Rennes – Bretagne Atlantique : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex
Centre de recherche INRIA Sophia Antipolis – Méditerranée : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399